

REMARKS

Claims 1, 2, 5, 6, 8, 9 and 10 have been amended. Support for the claim amendments can be found at page 6 of the specification. Claims 1-12 are pending and under consideration. Reconsideration is respectfully requested.

I. REJECTION OF CLAIMS 1-3, 5-7 AND 9-11 UNDER 35 U.S.C. § 103(a) AS BEING UNPATENTABLE OVER MEYER ET AL. (U.S. PATENT NO. 5,684,955; HEREAFTER “MEYER”)(newly cited) IN VIEW OF MARTIN (U.S. PATENT No. 6,438,746)(previously cited):

Claim 1 has been amended to recite “detecting that a certain class-type variable is contained in an execution statement to be executed in parallel or a certain class-type variable is specified in a parallelization directive as a class to be parallelized”.

At page 3 of the Office Action, the Examiner asserts that in Martin, the source code includes class-type variables (see column 7, lines 34-39).

The Applicants respectfully submit that Martin fails to disclose “a certain class-type variable is contained in an execution statement to be executed in parallel or a certain class-type variable is specified in a parallelization directive as a class to be parallelized,” as recited in amended claim 1. That is, Martin does not include any class-type variables (see FIG. 9 and column 7, lines 34-39).

Further, the Examiner asserts that Meyer discloses “class-type variables in the source code of an object-oriented program at column 5, lines 28-45 and that the source code includes calls or directives for distribution at column 6, lines 1-10.

Meyer merely discloses an object-oriented application which can be distributed over a plurality of operating system processes, and the programmer enters the calls for trans-process communication into the source of application. The calls are not calls or directives for distribution but instead are calls for messages sent from one process to another process. Thus, neither Meyer nor Martin, individually or combined, disclose “detecting that a certain class type variable is contained in an execution statement to be executed,” as recited in amended claim 1.

Further, the Examiner asserts that at column 9, lines 38-46, Meyer disclose stub objects comparable to the Applicants’ “object of the class” as recited in claim 1. However, at column 9, lines 38-46, Meyer discloses representative objects should be present as stub objects instead of real objects in a distributed application in operating system processes which do not themselves

contain a real object. The stub objects are then available locally as a contact point in the event of a request to the real object. Therefore, in Meyer, the stub object does not have the same function as the real object.

The Applicants respectfully submit that a class is a template that describes the data and behavior associated with instances of the class (see <http://java.sun.com/docs/books/tutorial/getStarted/application/classdef.html>). Thus, “an object of the class” and “an original object of the class” have the same function. In the present invention the same objects are instantiated for the parallel processing. That, is, the present invention discloses “generat[ing] said object in addition to an original object of the class” in claim 1. However, in Meyer, distribution of the objects is carried out and the representative object which is the stub object of the real object remotely instantiated, is necessary in the local computer.

Further, Meyer fails to disclose “generating an instruction to call construction instruction routine for an object of the class upon the detection, before said execution statement to be executed in parallel or an execution statement to be parallelized... and generating an instruction to call a destruction instruction routing for the generated object of the class upon the detection, after said execution statement to be executed in parallel or said execution statement to be parallelized, “ as recited in claim 1.

Instead, in column 10, lines 14-35 as pointed out by the Examiner, Meyer discloses SX_NEW instantiates a real object in another operating system process and a stub object in a local computer. Since the “stub object” does not correspond to the Applicants’ “object of the class” in claim 1, Meyer does not disclose “an instruction to call a construction instruction routine for an object of the class upon the detection,” as recited in claim 1. Nor does Meyer disclose “an instruction to call a destruction instruction routine for the generated object of the class,” as recited in claim 1.

Although the above comments are specifically directed to claim 1, it is respectfully submitted that the comments would be helpful in understanding differences of various other rejected claims over the cited references.

Based upon the comments above, the combination of Meyer and Martin fails to establish a *prima facie* case of obviousness over the present invention. Thus, it is respectfully submitted that the rejection is overcome.

II. REJECTION OF CLAIMS 4, 8 AND 12 UNDER 35 U.S.C. § 103(a) AS BEING UNPATENTABLE OVER MEYER IN VIEW OF MARTIN AND FURTHER IN VIEW OF

UMEKITA ET AL. (U.S. PATENT NO. 5,452,461; HEREINAFTER
"UMEKITA") (previously cited):

Claims 4, 8 and 12 depend from independent claims 1, 5 and 9, respectively. Therefore, the comments mentioned above in section I, may be applied here.

III. CONCLUSION:

In view of the foregoing amendments and remarks, it is respectfully submitted that each of the claims patentably distinguishes over the prior art, and therefore, defines allowable subject matter. A prompt and favorable reconsideration of the rejection along with an indication of allowability of all pending claims are therefore respectfully requested.

If there are any additional fees associated with filing of this Amendment, please charge the same to our Deposit Account No. 19-3935.

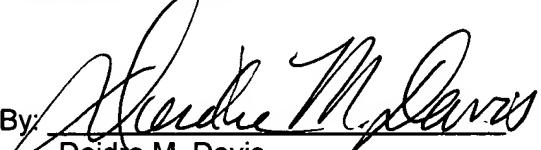
Respectfully submitted,

STAAS & HALSEY LLP

Date:

Oct. 3, 2005

By:


Deidre M. Davis

Registration No. 52,797

1201 New York Avenue, NW, Suite 700
Washington, D.C. 20005
Telephone: (202) 434-1500
Facsimile: (202) 434-1501